CLAIMS

We claim:

1. An apparatus comprising:

a data storage unit, of a cache, to speculatively provide data responsive to an access for an instruction;

circuitry coupled to said data storage unit to perform one or more arithmetic logic unit (ALU) functions specified by said instruction on the speculatively provided data;

hit/miss logic, of the cache, to determine if said access was a hit or a miss; and

a replay mechanism coupled to the hit/miss logic to replay said instruction if the access is determined to be a miss.

2. The apparatus of claim 1, wherein the replay mechanism includes a delay unit to delay a copy of the instruction for approximately a length of time required for the hit/miss logic to complete the determination.

3. The apparatus of claim 2, wherein the length of time includes time additionally required for the circuitry to complete the one or more arithmetic logic unit (ALU) functions.

4. The apparatus of claim 1, wherein the data storage unit and the circuitry is clocked at a faster frequency than the hit/miss logic.

5. The apparatus of claim 1, wherein said apparatus includes:

an execution subcore clocked at a faster frequency that said hit/miss logic, said execution subcore including said data storage unit and said circuitry.

6. A method comprising:

fetching an instruction;

accessing data from a cache location having stored therein data represented as possibly being associated with an address referenced by the instruction;

performing one or more arithmetic logic unit (ALU) functions specified by the instruction on the data;

determining that the data is not associated with the address referenced by the instruction, wherein the accessing is performed at a higher clock rate than the determining; and

replaying, responsive to the determining, the instruction and any other instructions that received results of that instruction.

7. The method of claim 6, wherein said performing is performed at a higher clock rate than the determining.

8. The method of claim 6, wherein said performing is performed at the higher clock rate.

9. The method of claim 6, wherein the replaying includes replaying a copy of the instruction that has been delayed for approximately a length of time required to complete the performing and the determining.

10. A system comprising:

a main memory including a plurality of addresses; and

a processor coupled to the main memory, including,

a cache memory having:

a data storage unit to store data represented as possibly being associated with a first address of the plurality of addresses, and

hit/miss logic clocked at a slower frequency than the data storage unit to determine whether the data is associated with the first address,

circuitry coupled to said data storage unit to perform arithmetic logic unit (ALU) functions, and

a replay mechanism coupled to the hit/miss logic to replay a first instruction referencing the first address in response to the hit/miss logic determining that the data was not associated with the first address.

11. The system of claim 10, wherein the replay mechanism also replays second instructions that received results of the first instruction in response to the hit/miss logic determining after the first instruction has been executed that the data was not associated with the first address.

12. The system of claim 10, wherein the replay mechanism includes a delay unit to delay a copy of the first instruction for approximately a length of time required to execute the first instruction and determine whether the data is associated with the first address.

13. The system of claim 10, wherein the hit/miss logic is clocked at a slower frequency than the circuitry.

14. A method comprising:

fetching an instruction that references a first address of a plurality of addresses included in a main memory;

accessing data from a cache's data storage unit, wherein the data is represented as being associated with one of said plurality of addresses;

performing one or more arithmetic logic unit (ALU) operations specified by the instruction on the data;

determining that the data is not associated with the first address, wherein the accessing is performed at a higher clock rate than the determining; and

replaying, responsive to the determining, the instruction and any other instructions that received results of that instruction.

15. The method of claim 14, wherein the replaying includes replaying a copy of the instruction that has been delayed for approximately a length of time required to complete the performing and the determining.

16. The method of claim 14, wherein the performing is performed at a higher clock rate than the determining.

17. A processor comprising:

a cache including,

a data storage unit having a plurality of cache locations, and

hit/miss logic;

arithmetic logic unit (ALU) circuitry to operate on data provided by said data storage unit prior to verification of cache hit/misses for said data by said hit/miss logic; and

a replay mechanism coupled to said hit/miss logic to replay instructions executed on data accessed from said data storage unit when such accesses are determined to be cache misses by said hit/miss logic.

18. The processor of claim 17, wherein the hit/miss logic is clocked at a slower frequency than said data storage unit.

19. The processor of claim 18, wherein a frequency at which the data storage unit is clocked is an integer multiple of the slower frequency.

20. The processor of claim 18, wherein the hit/miss logic is clocked at a slower frequency than said arithmetic logic unit (ALU) circuitry.

21. A method comprising:

fetching an instruction specifying an operation on data stored at an address;

accessing contents of a cache location based on said address;

executing said instruction based on said accessed contents;

determining whether said accessing resulted in a cache hit, wherein the accessing is performed at a faster clock frequency than the determining; and

if said accessing did not result in a cache hit, repeating said accessing, executing, and determining.

22. The method of claim 21, wherein the faster clock frequency is an integer multiple of a frequency at which the determining is performed.

23. The method of claim 21, wherein the executing is performed at a faster clock frequency than the determining.

24. The method of claim 21, wherein said executing is performed at the faster clock frequency.

25. A method comprising:

accessing data from a cache location having stored therein data represented as possibly being associated with an address referenced by an instruction;

performing one or more arithmetic logic unit (ALU) functions specified by the instruction on the data;

determining that the data is not associated with the address referenced by the instruction; and

replaying, responsive to the determining, the instruction and any other instructions that received results of that instruction.

26. The method of claim 25, wherein the replaying includes replaying a copy of the instruction that has been delayed for approximately a length of time required to complete the performing and the determining.

27. The method of claim 25, wherein the accessing and the performing are performed at a higher clock rate than the determining.